Playtesting and Iterative Design

The Most Important Process for Making Great Games

Ira Fay, *Hampshire College, Amherst, Massachusetts, U.S., ira@irafay.com*

Key Summary Points

1 Playtesting is the act of observing someone play your game. It is a valuable process for any game designer because it provides true insight into the game, allowing you to improve it.

2 For learning games, playtesting also focuses on understanding if the player is actually learning anything from playing your game and ideally how well that knowledge is retained and/or transferred.

3 The iterative development process is a three-step loop (design, build, test) and playtesting is one of the three steps in that loop.

Key Terms

Playtesting
Iterative design
User testing
Design build test loop
Game development best practices
Playtesting for learning games

Introduction

Many people want to make a great game that engages players and provides educational impact, but how does one actually accomplish this goal? One process that can improve your chance of success is the iterative development process. This process is a repeating loop: first we design something, then we build it, and then we test it. Based on the results, we change our design and modify what we had built. Then we test the game again, modify the design based on the test, update the game accordingly, etc. To maximize quality, we strive to repeat these three steps (design-build-test) as many times as possible in the time available.

When we build something, it could be a very basic prototype (analog or digital), a specific feature in the game, or the final polish on a nearly finished game. At the start of development, the quality of our design is least certain, since we have not been able to test the game yet. If we spend a long time building the full game before testing, we will likely discover that our design needs to change and we have wasted a lot of time building unnecessary features. Therefore, especially near the beginning of development, it is best to build something small (such as a prototype), which can be tested quickly to confirm the design is on the right track.

When we talk about testing in the context of game development, we are really talking about playtesting. Playtesting is the act of observing someone play your game with the intention of understanding that player's experience. This is different than simply watching someone else play a game. While playtesting, we strive to truly understand what the player is thinking, feeling, and doing, and why. We then use that understanding to improve the game.

The ability to accurately and insightfully observe players is a skill that anyone can develop with practice. Every time you moderate a playtest, your observational skills will improve. Fortunately, we also have many tools available to support that effort, such as cameras or touch/click tracking software.

A playtest involves three people or groups:

1. Player
2. Moderator
3. Development team

The player plays the game while the moderator observes and takes notes. The moderator should attempt to track the player's actions (e.g., clicking, touching, choices), as well as more subtle things like pauses, points of confusion or excitement, or facial expressions. The goal of the playtest is to understand the player's actions and feelings, allowing the development team to improve the experience for other players in the future.

The moderator spends most of his or her time silently observing the player. Sometimes, the moderator may ask a question to understand what the player is thinking at a particular moment. If the player gets particularly stuck, the moderator may also choose to intervene, allowing the player to reach other areas of the game. To maximize the player's comfort and willingness to speak openly about the game, often the development team will also observe in a separate area, possibly via a one-way mirror or recordings afterward. Sometimes a member of the development team may serve as the moderator, but an unaffiliated person as the moderator may minimize bias and ensure that the player feels comfortable critiquing the game honestly.

At the earliest stages of development, when the game is still far from being finished, it is sometimes easy and efficient for a single person to be the player, the moderator, and a member of the development team! In a certain state of mind, a person can play a game while simultaneously observing themselves play the game, almost like an out-of-body experience. After the playtest, the person can write notes about the experience and share them with the team. Ideally, such notes will be analyzed, leading to improvements to the design and continued progress around the design-build-test loop.

For learning games, the playtesting process can be even more complicated. Not only are you trying to observe levels of fun and how effectively the player can control the game, but you are also trying to understand how much the player learns by playing the game. First you will need to identify what you want the player to learn. For simple concepts, such as a fact or piece of trivia, often this learning is relatively binary—they either know it or not. Whether the player learned the fact can be assessed through simple questions from the moderator, or through observation of in-game activities. For more complicated concepts, such as understanding complex system interactions or almost any topic beyond simple facts, you will likely need to follow up with players after they complete the game playing experience. Though playtesting is not a science, there are some processes that you can follow as a moderator and playtest designer to maximize your chances of gathering useful information that is as unbiased as possible.

In this chapter, I will discuss the iterative design loop (design-build-test), best practices in playtesting, and how designers of games for educational impact can modify their playtesting practices to support their specialized goals.

Case Study One: *Unearthed*

Figure 1: Sample screenshot of Unearthed.

Unearthed was designed to help middle school students learn grammar, and very specifically, subject-verb agreement. The team for *Unearthed* consisted of five or six experienced game developers, who have experience in engineering, art, production, design, audio, and quality assurance. The design goal of *Unearthed* was to help middle school students learn grammar, and very specifically, subject-verb agreement. Early in the design process I met with educational experts and middle school teachers, discovering how the subject is best taught in classrooms. Several people suggested the idea of half-sentence matching. Students would see half-sentences with subjects and half-sentences with verbs, and then need to match the sentences properly. With that, we had the core idea for our game. We created an early prototype and we quickly identified several questions:

1. What is the best way for students to connect two half sentences on a tablet device? Touch one, and then touch another? Touch and drag? Multitouch?
2. Should the full sentences make sense logically, or is it sufficient to accept gibberish sentences as long as the subject and verb agree?
3. Are students actually learning anything by playing the game?

As I mentioned above, whenever I playtest I strive to create questions in advance that are specific to my current design needs. I also create some general questions (likely quantitative) that I ask at every playtest, allowing the team to identify any trends that emerge. Finally, I also leave some room for the moderator to improvise questions as needed.

Below are some examples of the questions I asked at this stage of playtesting *Unearthed*. Remember that the moderator strives to answer these questions himself or herself, based on observation of the player. These questions can likely be answered through observation, but sometimes the moderator must ask the player a question to be more certain of the answer. Of course, when talking to someone who is playing a game, the player's brain is occupied and he or she may have difficulty playing the game and talking to you at the same. Therefore, if you must ask the player a question during gameplay, try to keep it short and easy to answer.

A.   Does the player understand the goal of the game?
B.   If so, how does the player naturally try to connect the half-sentences?
C.   Does the player seem to understand the correct answers, or is the player often guessing at the right answer?
D.   Does the player seem to improve over time?

Along with the sample questions for the moderator above, we also surveyed the player:

1.   Demographic information (e.g., age, gender)
2.   Rate the game on a scale of 1-5.
3.   Rate the game on a scale of too easy/just right/too hard.
4.   Name a few things that could be improved and a few things you liked. Any other comments?

Questions (A) and (B) focused on basic usability issues, while questions (C) and (D) focused on learning objectives. The (1) - (4) survey questions considered whether the player was having fun, though a better indication can be gauged by the moderator during the playtest by allowing the player to stop playing at any time, and seeing when the player stops. Fun and engagement can be inferred by how long the player continues to play.

As it turns out, regarding question (B), players tried to connect half-sentences in all three ways (touch-touch, touch-drag, multi-touch). Therefore, we changed the input detection code to support all three methods.

Regarding question (C), we noticed that many players were guessing very quickly. We had not implemented a major penalty for guessing, nor a reward for getting many right in a row. Therefore, since the game was timed to last 60 seconds, the optimal strategy for any player was to guess immediately, without taking the time to read the sentences at all. Based on our playtesting, we changed the scoring and messaging to strongly reward correct answers in a row, and we also penalized wrong answers.

For question (D), we did see some high scores increasing over time, but we cannot be sure if that was caused by actual learning, or if it was something else like a better understanding of the game controls. To get a clearer answer to this question, we would need to do a rigorous study.

Key Frameworks

I believe there is widespread consensus among professional game developers that iterative development is a wise process and that playtesting is a useful tool for designers to employ. Playtesting allows developers to understand how players will experience their game, leading to increased fun, sales, and a higher quality final product.

Therefore, instead of asking *if* we should playtest and iteratively develop, the more interesting question is *how* we effectively playtest and iteratively develop. Fortunately, with practice and care, we can improve our processes and our playtesting skills.

Particularly for learning games, it is important to be clear in your own mind (and your boss's mind!) that playtesting is not intended to be a scientific process from which you can publish statistically significant results. Playtesting is intended to be a tool for the development team, allowing the team to iteratively improve the game. While the moderator or playtest designer may attempt to assess whether the player has learned anything by playing the game, those results need not be scientific, they merely need to be useful enough to guide development. The time, effort, and cost required to create a well-designed and scientifically rigorous assessment study simply is not practical during most development cycles. Of course, if you have the time and money to do so, great! Such rigorous, scientific studies of learning games will help all developer in the long run. But for most of us, nonscientific playtesting can suffice.

As a quick introduction to moderating a playtest for those who are unfamiliar with it, I offer the following steps that I typically follow:

1. As moderator, work with the development team to craft the questions that you will try to answer in the playtest, based on the team's current stage of development. Early in development, the team might be wondering about basic game controls or the story hook. Later in development, the development team might want to know if level seven is taking within the desired 120-180 seconds to complete. Either way, create a list of questions that you will try to answer during the playtest.

2. After you have a list of questions, recruit players for the playtest. The players should ideally match the target demographic of the game, and have an appropriate level of experience with the game given the questions that are being asked. For example, if you are trying to understand the effectiveness of the tutorial, it would be best to test with players who have never played before. On the other hand, if you are trying to understand if the crafting system has sufficient depth to retain players for ten or more hours, playtest with experienced players.

3. Before the playtest session, create a script of events with approximate times. What do you want the player to do first? What questions will you try to answer during that time? What will the player do next? How long do you want to spend with each player? A typical playtest lasts 30-60 minutes, though it can vary widely based on your needs. Also, before

the playtest starts, it is wise to ensure your recording devices are working properly, if you intend to use them. I often prepare an online survey for the player to complete just after finishing the session.

4. At the beginning of your session with a player, remind the player that he or she can stop at any time. Not only is this ethical, it is also a very useful measure of engagement. If players want to stop at certain points of the game, such behavior provides a helpful clue to the development team. Also remind the player that the game is unfinished, and that you are testing the game, not the player. It is important to create an emotionally comfortable environment for the player so that he or she is more willing to give you honest feedback.

5. During the session, it is generally best to stay quiet and focus on observing the player. Ideally, you want the player to forget you are even present, allowing you to witness a more authentic game playing experience. If the player asks you a question, take note of it and answer at the end of the session.

6. Once the playtest session is done, finish any necessary notes to yourself while the events are still fresh in your mind. Give the player a survey, if desired. Prepare to welcome your next playtester.

The steps above provide a rough outline to moderating a playtest, but you should modify them as needed to match your preferences. As a note, there are many existing and comprehensive books on usability testing, which serve as a foundation for playtesting (please see the Resources section).

Case Study Two: *The Tomes*

The Tomes is a game designed to help middle school students learn vocabulary. After talking with educational experts, we learned that one important facet of long-term vocabulary retention is using and seeing the words in context, not merely using flash cards. Therefore, we considered a game design that supported this learning objective and encouraged students to read and use vocabulary in context. We settled on the idea of a choose-your-own-adventure game, with a graphic novel visual style.

We wanted the game to appeal equally to boys and girls, but due to the complexities of writing a work of interactive fiction, we decided to restrict players to a male protagonist. When we playtested the game, we gathered basic quantitative data from players in the form of a very simple survey, similar to the one listed above. Below are the results from the question, "How fun was the game? (1 = worst, 10 = best):"

Three boys, average 8.7
Three girls, average 4.8

While there were other variables and considerations, this result stood out. Even though the sample size is so small, the result is still useful.

When we saw that playtest result, we revisited the idea of giving the player a choice of protagonist gender. It would mean a bit of rewriting and rethinking the story, plus additional art assets, but we decided it was worth a try. This was a classic case of the design-build-test loop:

1. We designed a game with a male-only protagonist.
2. We built the original prototype with a male-only protagonist.
3. We playtested the prototype and gathered observations from real players.
4. We analyzed the results and decided to modify the design, adding the choice of protagonist gender.
5. We updated the game by allowing players to choose a protagonist gender.
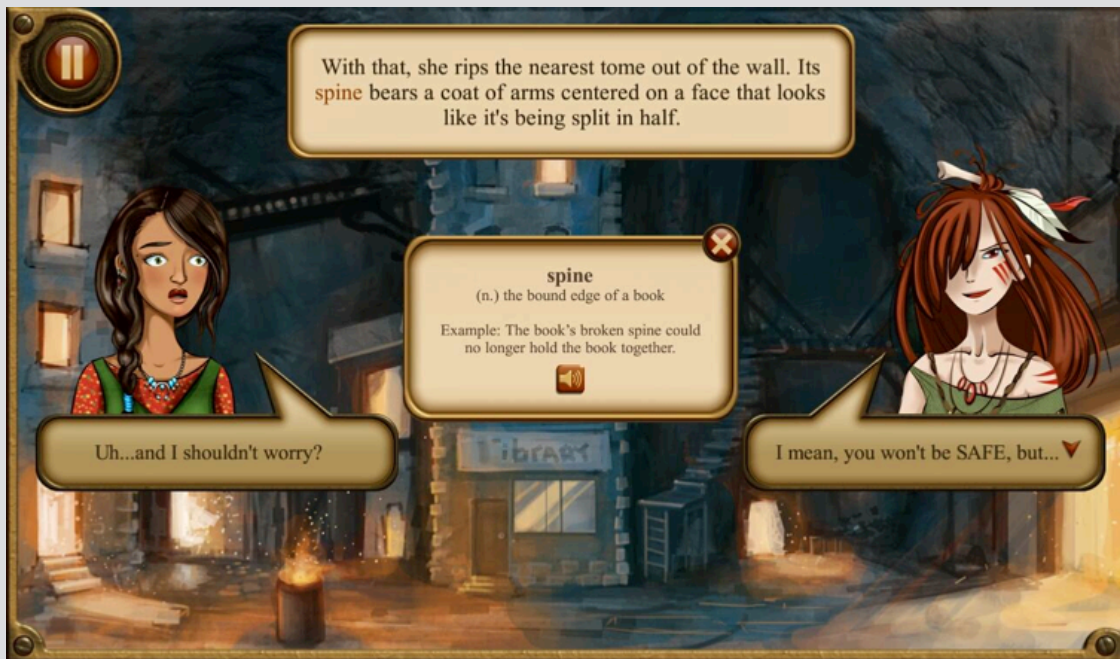6. We playtested the new version and gathered observations again.

Figure 2: *The Tomes* sample screenshot, having touched a vocabulary word.

Below are the results we got when we playtested again with gender choice and the exact same survey question:

Three boys, average 9.0
Three girls, average 8.3
(Note: these were different playtesters from the previous version)

Of course, between the two versions, other things may have changed too. The art was more polished, the writing had improved, and ten other uncontrolled factors may have changed. At this point in development, we were reasonably sure that the gender choice had a positive impact, but either way, the average rating of fun increased for girls. We continued to track the numbers over the course of the project, and if we saw dips in the future we would try other experiments as well. As it turns out, the numbers stayed high and we kept the gender choice in the game.

Figure 3: Once one touch away from starting a game of *Food Web.*

Assessment Considerations

For playtesting

As mentioned above, playtesting does not often result in scientific, statistically significant results, which is fine. Instead, playtesting is intended to provide useful guidance and feedback to the development team. When assessing the effectiveness and value of your playtesting, consider the following questions:

1. **Are you playtesting with your target audience?** If your game is designed for sixth to eighth graders in the United States and you playtest with older kids, your data may not be as effective. While it can be useful to playtest with anyone at all, strive to playtest with your target audience.
2. **Are you reusing playtesters from session to session?** This may seem obvious, but sometimes there is a shortage of playtesters and you must reuse playtesters. In that case, any information you are trying to glean related to their first-time play experience may not be useful. Instead, focus on the more advanced aspects of the game with playtesters who have seen an earlier iteration.
3. **With similar methodology, do you see changes over time?** Ideally, you will establish a playtesting process early in your development cycle, which you can then use whenever you playtest. For example, you could ask the playtester to complete a survey, and then keep some of the questions consistent from session to session. Over time, you may see trends in the results that can inform design.

For iterative development

There are a variety of software development methodologies that strive to facilitate the iterative development process. For example, all the flavors of agile development are, at their core, trying to help teams employ iterative development. A thorough analysis of all iterative development methodologies is beyond the scope of this chapter, but here are some things to consider when thinking about your development process:

1. Is there effective communication flowing between members of the same discipline? Across disciplines? At different levels of seniority and management? Between client and publisher?
2. Does the team regularly reflect on its current process and strive to improve it?
3. Does everyone on the team remember the importance of the design-build-test loop, and act accordingly?
4. Is the schedule and budget reasonable to support the goals of the project? Since a precise schedule is very difficult to know at the beginning of the project, are all key stakeholders aware of the design-build-test loop?
5. Does the team playtest regularly?
6. Is it easy for anyone on the team to get a current version of the game and play it, even (especially!) in the middle of development?

Future Needs

There are many passionate people interested in improving educational systems and/or making fun games. The reality is that great teaching is quite difficult, making fun games is quite difficult, and doing both at the same time is even harder. As this aspect of the games industry continues to develop, veteran game developers need to partner with veteran teachers to craft experiences that take the best of both crafts.

A scientific study that demonstrates the value of playtesting and iterative development may be useful, but may not be necessary to show their effectiveness. Anyone who has ever made a game and moderated a playtest knows that playtesting is a valuable tool. Seeing in advance what your players think of your game is extremely beneficial.

Case Study Three: *Food Web*

At the time of this writing, I recently finished development on a science game designed to help middle school students learn about the food chain or food web. Specifically, the game strives to help students learn that predators eat prey and organisms need food to survive.

The game design was inherited from a different team and at the beginning of the project all the key stakeholders agreed on these four learning objectives:

1. A food web ecosystem exists.
2. Organisms have predators and preys.
3. Specific predator-prey relationships exist in a rainforest habitat. For example, fig seeds are eaten by Black Rail birds, which are eaten by South American Bushmaster Snakes, which are eaten by Southern Crested Caracara birds.
4. Organisms eat other organisms for energy.

As we were nearing the end of development, we had the opportunity to playtest. I created many questions for the moderator to strive to answer, based on the learning objectives and various lingering design questions. Here are a few example questions:

1. When players start the app for the very first time, do they touch the unlocked animal and then start the game, or do they get lost in the menu system?
2. Do first-time players understand that they need to touch next to their animal to move?
3. Do players quickly learn that predators will eat them?
4. Do players quickly learn that they need to eat prey?
5. Do players learn over time that they spend energy to move, and gain energy when eating prey?
6. Do players learn that there are some animals that are neither prey nor predators, and that those animals cannot be eaten?
7. Do players ever unlock a new animal by going to the Upgrade menu?
8. Do players realize that different animals have different predators and prey?
9. Do players notice the goals?
10. Do the goals drive player behavior in some way? (e.g., they notice they need to eat mosquitoes, so they change to an animal that eats mosquitoes?)
11. After a while, can players name any specific predator-prey relationships? (e.g., Agoutis eat Fig Seeds. Bushmaster Snakes eat Agoutis.)

As you can see, there are many questions here related to usability (e.g., Can players start the game?, Control their animal?, Notice goal messaging?, Go to the Upgrade menu?). There are also quite a few goals specifically related to the learning objectives. When an observant and caring moderator takes time to watch players play the game, the moderator will be able to answer questions like the ones above. Such answers will be able to inform design.

In this particular case, we got positive results related to the learning objectives. For example, we knew players were understanding predator-prey relationships and naming specific animals because we had videos of players saying the animal names and relationships. Fortunately, we also caught some lingering issues related to the first-time player experience. New players would get lost in an interactive food web, instead of getting into playing the game first. We improved messaging, removed superfluous buttons, and reduced the number of screens shown to a first-time player before she or he started playing, all based on the playtest feedback. While the game is still not perfect, the informal and nonscientific playtesting helped us make good design choices and improve the game during development.

Best Practices

When considering how to playtest, here are some guidelines to keep in mind:

1. **Test early, test often:** A playtest with a sample size of even one or two people is vastly superior to no playtesting at all. Also, the more frequently you playtest, the fewer things will have changed between playtests, which makes it easier to determine cause and effect.
2. **Create a good environment for observation:** The foundation of an effective playtest is carefully observing the player. Strive to create an environment where the moderator (and supporting tools such as cameras and/or click/touch tracking software) can do their job. For example, it is often better to observe a single player carefully, instead of many players simultaneously. In the many simultaneous players situation, you will get more survey data, but you will miss detailed observations of individual player actions. In the single player situation, you will be able to focus on everything the player does, which will allow you to better deduce what is happening inside that player's mind.
3. **Practice moderating playtests:** With practice, playtest moderators will get better at observing players, noticing subtle things, asking open-ended questions, staying patient, and taking good notes during and after the session. It is a skill that can be improved, so practice it.
4. **Create a good environment for feedback from the player:** A moderator can observe quite a few things, verbally and non-verbally, but it is very difficult to get at the player's thoughts. Often players will not be able to accurately communicate their own experiences. To attempt to understand a player's experiences more fully, moderators can ask direct questions to the player during or after gameplay. Cultivate a welcoming environment that exudes serious curiosity about the player's ideas is something that can be cultivated over time.

5. **Create quantitative questions that you can ask consistently throughout development:** The process will not be rigorously scientific, but you will have a few questions and answers that you can track, which will be useful for observing possible trends.
6. **Create qualitative questions as needed to support your current design questions:** If you take time before a playtest to think about your current design questions, it will be relatively easy to create questions to match. With a list of questions to answer, the moderator can help gather the information you need.

Resources

Brian Schrank (http://www.brianschrank.com/capstone/resources/Playtesting_reports_template.pdf).

Holly Gramazio, the lead game designer at Hide & Seek (http://hideandseek.net/2011/01/26/a-guide-to-playtesting-from-h-g-wells/).

Best Practices: Five Tips for Better Playtesting, by Vin St. John (http://gamasutra.com/view/feature/185258/best_practices_five_tips_for_.php). Notice in the comments that Vin responds to the first comment, presumably to a random person on the internet who he does not know, "We're constantly trying to improve our process, so if you have any criticisms or suggestions I would welcome them."

Finding Out What They Think: A Rough Primer To User Research, by Ben Lewis-Evans (http://www.gamasutra.com/view/feature/169069/finding_out_what_they_think_a_.php, http://www.gamasutra.com/view/feature/170332/finding_out_what_they_think_a_.php).

References

Brathwaite, B. & Schreiber, I. (2008). *Challenges for game designers: non-digital exercises for video game designers*. Cengage Learning.

Fay, I. (2011). Filtering feedback, in G. Costikyan & D. Davidson (Eds.) *Tabletop: Analog Game Design,* Pittsburgh: ETC Press.

Gramazio, H. (2011). *A guide to playtesting from H.G. Wells,* accessed at: http://hideandseek.net/2011/01/26/a-guide-to-playtesting-from-h-g-wells/

Krug, S. (2014). *Don't make me think. Third edition.* New Riders.

Lewis-Evans, B. (2012). Finding out what they think: A rough primer to user research, accessed at: http://www.gamasutra.com/view/feature/169069/finding_out_what_they_think_a_.php, http://www.gamasutra.com/view/feature/170332/finding_out_what_they_think_a_.php

Norman, D. (2002). *The design of everyday things.* Basic Books.

Ries, E. (2011). *The lean startup.* Crown Business.

Schell, J. (2008). *The art of game design: A book of lenses.* CRC Press.

Schrank, B. *Capstone winter quarter playtesting guide,* accessed at: http://www.brianschrank.com/capstone/resources/Playtesting_reports_template.pdf

St. John, V. (2013). Best practices: Five tips for better playtesting, http://gamasutra.com/view/feature/185258/best_practices_five_tips_for_.php